We are IntechOpen,
the world's leading publisher of Open Access books
Built by scientists, for scientists

5,200
Open access books available

128,000
International authors and editors

150M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Multiresolution Approaches for Edge Detection and Classification Based on Discrete Wavelet Transform

Guillermo Palacios, J. Ramón Beltrán and Raquel Lacuesta
University of Zaragoza
Spain

1. Introduction

Detecting edges is a very well known subject in the image-processing field. Edge detection is the process of the localization of significant discontinuities in the grey level image and the identification of the physical phenomena that originated them. Those significant intensity changes occur at different resolution or scales for a given image. As suggested by Rosenfeld and Thurston (Rosenfeld & Thurston, 1976) and Marr (Marr, 1982), we can obtain a description of the image changes at different scales combining the information given by an edge detector applied at different resolutions. This is the aim of the work presented in this paper.

The first aspect to be covered by multiresolution analysis is the filter chosen to accomplish the low-pass filtering of the image at different scales. At a single resolution, low pass filtering is imposed because differentiation is an ill-posed problem (Torre & Poggio, 1984). The needed regularization process is implemented by means of a low pass filter. Marr (Marr, 1982) proposed the Gaussian filter because its optimal behaviour in terms of the smoothing and the localization in both the spatial and frequency domains. This filter has been commonly used in edge detectors. In a multiresolution approach the first or second directional derivatives of the low pass filtered image with Gaussians of different widths are used to detect edges. In the Bergholm edge focusing method (Bergholm, 1987) various edge maps extracted at different scales are integrated allowing distinguishing shadows contours from perfect ones using Canny's operator (Canny, 1986) with different widths. Another possibility is to describe the image in terms of the scale space as proposed by Witkin (Witkin, 1983) and to detect edges in terms of the zero crossing of the Laplacian operator with different widths (Park et al., 1995) (Eklundth et al., 1982).

Other multiresolution methods have been proposed. Mallat and Zhong (Mallat & Zhong, 1992) related multiscale edge detection with the discrete wavelet transform (DWT). They proposed a wavelet to perform edge detection and they showed that the evolution of wavelet local maxima across scales characterizes the shape of irregular structures. In our work we will use the wavelet-based algorithm proposed by Mallat and Zhong and we will show the condition that must be satisfied by the Gaussian filter to be comparable with the Mallat and Zhong's wavelet. Our aim is to detect and classify different edge types. Various edge profiles have been proposed. Rosenfeld (Rosenfeld & Kak, 1976) proposed the step,

ramp, pulse and stair as the basic edge types. He stated that these profiles are suited for a first intuitive classification of the edges found in the contours of real images.

William and Shah (Williams & Shah, 1990, 1993) have studied these edge types using the first directional derivative of the Gaussian. Some other profiles like the blurred step have also been proposed and analyzed (Ziou & Tabbone, 1993). In a first step of this work we are going to analyze the evolution of the modulus of the wavelet coefficients at the edge position in order to classify the edges into four different profiles: step, ramp, stair and pulse (Beltrán et al., 1994). Then we will propose a general schema to detect, analyze and classify different edge types.

Due to the high pass filtering operation involved in the edge detection algorithms, the noise is always amplified when detecting edges. Thresholding has been the most common way to eliminate the irrelevant or noise detected edges (Canny, 1986; Marr, 1982). To validate the proposed classification schema we will present the noise (Gaussian noise) as a new edge class (Beltrán et al., 1998). Then, we have analyzed this contour type and we have modified the proposed classification algorithm to include the noise as a new edge type. With the noise edges labelled we can easily implement a noise-filtering algorithm. Finally, we have developed a new classification algorithm including other edge profile models, such as the roof, ridge and two non-antisymmetrical step profiles, like the ones proposed by Paillou (Paillou, 1994).

This chapter is divided as follows. Section 2 presents a survey of the wavelet formalism introduced by Mallat and Zhong. Then, we will show the geometrical contour types: step, ramp, pulse and stair. Section 3 presents the edge detection algorithm including the wavelet algorithm implementation. Section 4 presents the classification algorithm implementation details as well as the classification results obtained processing a 256x256 grey level synthetic image with four objects: a circle, a square, a triangle and a narrow line, each one having a different contour type. Section 5 deals with the characterization of the noise as a new type of contour. Section 6 copes with the new contour types just introduced, the modified classification algorithm together with the obtained results. In section 7 we present the main conclusions and the future work.

2. Theoretical basis

Mallat and Zhong (Mallat & Zhong, 1992) introduced the relationship between the wavelet transform and a multiresolution edge detection algorithm. We are going to briefly summarize those results. Let be $f(x,y) \in L^2(R^2)$ an image and $\psi(x,y)$ a wavelet. The bidimensional wavelet transform of $f(x,y)$ is defined as:

$$W_s f(u,v) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(x,y)\frac{1}{s}\psi\left(\frac{x-u}{s},\frac{y-v}{s}\right) \tag{1}$$

If we define the dilation by a factor s as:

$$\psi_s(x,y) = \frac{1}{s}\psi\left(\frac{x}{s},\frac{y}{s}\right) \tag{2}$$

and $\tilde{\psi}(x,y) = \tilde{\psi}(-x,-y)$, we can rewrite (1) as a convolution:

$$W_s f(u,v) = f * \tilde{\psi}_s(u,v) \tag{3}$$

So, as expressed in (3), the wavelet transform can be seen as the filtering of f(x,y) by the filter $\tilde{\psi}_S$ (x,y), that is a variable width bandpass filter (Mallat, 1989). It is possible to define N directional wavelets ψ^i(x,y) ($1 \le i \le N$), satisfying energy conservation properties. In such a case the directional wavelet transform of f(x,y) is defined as:

$$W_S^i f\left(u,v\right) = f * \tilde{\psi}_S^i\left(u,v\right) \tag{4}$$

Equation (4) represents the filtering of f(x,y) by the bidimensional, directional and bandpass filter $\tilde{\psi}_S^i$ (x,y).

We can define the Discrete Wavelet Transform (DWT) by selecting the scales inside a dyadic grid; that is to say, the scale could be expressed as S=2^j with $j \in$ **Z**. Therefore, for discrete signals, we can understand the 2-D wavelet transform as the result of filtering the 2-D signal (the original image) with a bandpass directional FIR filter.

Mallat and Zhong (Mallat & Zhong, 1992) designed a function specially suited for edge detection purposes, which is a wavelet. This function is not a orthogonal wavelet, so the only condition to be satisfied by ψ(x,y) is:

$$\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} \psi\left(u,v\right)dudv = 0 \tag{5}$$

We can define two functions ψ^1(x,y) and ψ^2(x,y) as:

$$\psi^1(x,y) = \frac{\partial\theta(x,y)}{\partial x}, \quad \psi^2(x,y) = \frac{\partial\theta(x,y)}{\partial y} \tag{6}$$

Where θ(x,y) is a smoothing function, that is to say, the integral over x and y is one, and it converges to zero at infinity. With these conditions is easy to show that both ψ^1(x,y) and ψ^2(x,y) satisfy (5), so they are wavelets. Following equation (3) we can say that the wavelet transform of f(x,y) is:

$$\begin{aligned} W_S^1 f(x,y) = f * \tilde{\psi}_S^1(x,y) \\ W_S^2 f(x,y) = f * \tilde{\psi}_S^2(x,y) \end{aligned} \tag{7}$$

From (6) and (7) it can be shown that the wavelet transform is the gradient of the image smoothed by a factor or scale s. It can be expressed as:

$$\begin{pmatrix} W_s^1 f(x,y) \\ W_s^2 f(x,y) \end{pmatrix} = s \begin{pmatrix} \dfrac{\partial}{\partial x}(f * \theta_s) \\ \dfrac{\partial}{\partial y}(f * \theta_s) \end{pmatrix} = s\vec{\nabla}(f * \theta_s) \tag{8}$$

We can define the modulus **M** and the phase **Φ** of the gradient at scale s as:

$$M\left(\vec{\nabla}(f * \theta_s)\right) = \sqrt{\left|W_s^1 f(x,y)\right|^2 + \left|W_s^2 f(x,y)\right|^2} \tag{9}$$

$$\Phi\left(\vec{\nabla}(f * \theta_s)\right) = \mathrm{atan}\left(\frac{W_s^2 f(x,y)}{W_s^1 f(x,y)}\right) \tag{10}$$

A point (x_0, y_0) will be an edge point of the smoothed image $f * \theta_s (x,y)$ if in this point there is a relative maximum of **M** in the direction addressed by **Φ**. The above statement is the classical definition of edge detection proposed by Canny [5]. In the wavelet case we have a discrete set of scales $s = 2^j$ $(0 \le j \le J)$, and we can calculate the edges at each scale and not at an only one scale as described in Canny's work. So, we can conclude that we can implement a multiresolution edge detection algorithm by means of the 2-D wavelet transform. The smoothing function could be a new defined function, as in Mallat and Zhong's work, or a Gaussian function, as in Canny's work.

Mallat and Zhong (Mallat & Zhong, 1992) defined θ(x,y) as a separable bidimensional cubic spline, so ψ(x,y) is a separable bidimensional quadratic spline. Their 1-D expressions are:

$$\hat{\theta}(x) = \begin{cases} \frac{8}{3}x^3 + 8x^2 + 8x + \frac{8}{3} & \text{if } -1 \le x \le -1/2 \\ -8x^3 - 8x^2 + \frac{4}{3} & \text{if } -1/2 \le x \le 0 \\ 8x^3 - 8x^2 + \frac{4}{3} & \text{if } 0 \le x \le 1/2 \\ -\frac{8}{3}x^3 + 8x^2 - 8x + \frac{8}{3} & \text{if } 1/2 \le x \le 1 \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

$$\hat{\psi}(x) = \begin{cases} 8x^2 + 16x + 8 & \text{if } -1 \le x \le -1/2 \\ -24x^2 - 16x & \text{if } -1/2 \le x \le 0 \\ 24x^2 - 16x & \text{if } 0 \le x \le 1/2 \\ -8x^2 + 16x - 8 & \text{if } 1/2 \le x \le 1 \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

In the 2-D case we can define $\theta(x,y) = \hat{\theta}(x)\hat{\theta}(y)$ and $\psi(x,y) = \hat{\psi}(x)\hat{\psi}(y)$. We can compare the wavelet and the Gaussian functions behaviour. In figure 1 we can observe that both functions present a similar aspect. We have obtained a relationship between the Gaussian width σ and the scale s of the wavelet analysis. The Gaussian normalization constant is:

$N = \dfrac{1}{\sqrt{2\pi} \cdot \sigma}$, and the smoothing function value at the origin is $\theta(0) = \dfrac{4}{3}$.

Equating both quantities we obtain σ= 0.3. So, to obtain a similar analysis with the wavelet and the Gaussian function, the Gaussian width and the scale should be related by the aforementioned expression.

Now we will to present the edge profiles considered initially. We have considered that each profile contour is a function that represents a change in the grey level with respect to the image dimensions. In this work we will name U_i the contour height, ω the contour width, x_0 the contour position (at the middle of the contour) and s the scale.
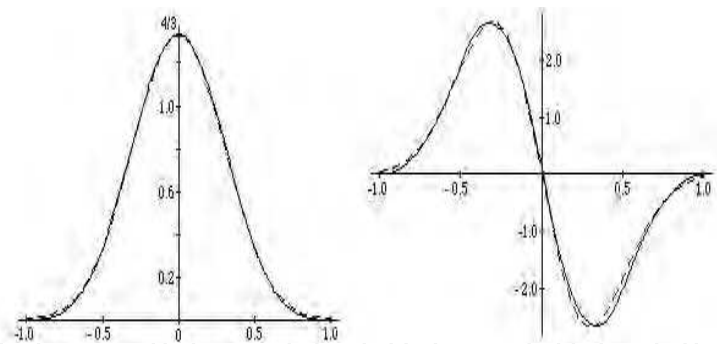
Fig. 1. Left: Solid line: Smoothing function. Dotted line: Gaussian with σ = 0.3. Right: Solid line: Wavelet. Dotted line: First derivative of Gaussian with σ = 0.3.

2.1 Step

A step profile is shown in figure 2. We have named x_0 the step location and U_0 the step height. Let be u(x) the step function.

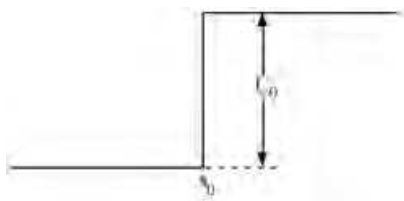$$u(x) = \begin{cases} U_0 & \text{if } x \geq x_0 \\ 0 & \text{if } x < x_0 \end{cases} \tag{13}$$

Fig. 2. Step profile. Horizontal axis represents pixels. Vertical axis represents grey level.

2.2 Ramp

The ramp profile is drawn in figure 3. The point x_0 is the middle point of the ramp; U_0 is the height and ω is the ramp width. The ramp slope is m = U_0/ω. Let be r(x) the ramp function.

$$r(x) = \begin{cases} 0 & \text{if } x < x_0 - \dfrac{\omega}{2} \\ mx + r_0 & \text{if } x_0 - \dfrac{\omega}{2} \leq x \leq x_0 + \dfrac{\omega}{2} \\ U_0 & \text{if } x > x_0 + \dfrac{\omega}{2} \end{cases} \tag{14}$$
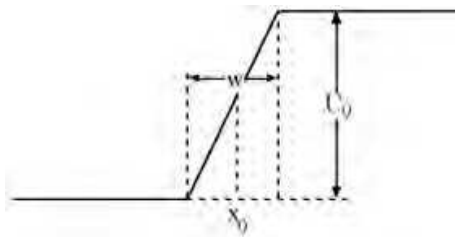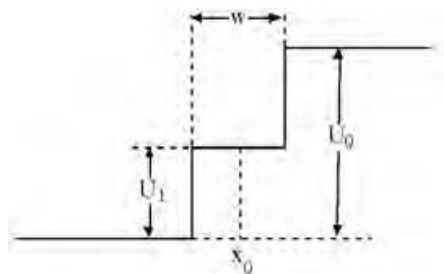

Fig. 3. Ramp profile. Horizontal axis represents pixels. Vertical axis represents grey level.

2.3 Stair

A stair profile with two steps, named s(x), is shown in figure 4. We have named the point x_0 the middle of the stair, ω is the stair width and U_0 and U_1 are, respectively, the steps height.

$$s(x) = \begin{cases} 0 & \text{if } x < x_0 - \dfrac{\omega}{2} \\ U_1 & \text{if } x_0 - \dfrac{\omega}{2} \le x \le x_0 + \dfrac{\omega}{2} \\ U_0 & \text{if } x > x_0 + \dfrac{\omega}{2} \end{cases} \tag{15}$$

Fig. 4. Stair profile. Horizontal axis represents pixels. Vertical axis represents grey level.

2.4 Pulse

A pulse profile, named p(x), is shown in figure 5. The point x_0 is the middle of the pulse, ω is the width. U_0 and U_1 are, respectively, the maximum and minimum height.

$$p(x) = \begin{cases} 0 & \text{if } x < x_0 - \dfrac{\omega}{2} \\ U_0 & \text{if } x_0 - \dfrac{\omega}{2} \le x \le x_0 + \dfrac{\omega}{2} \\ U_1 & \text{if } x > x_0 + \dfrac{\omega}{2} \end{cases} \tag{16}$$
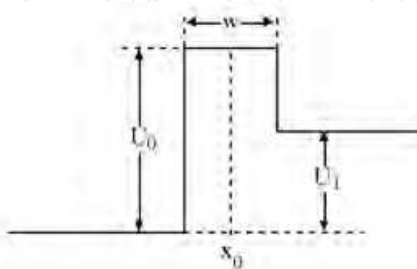

Fig. 5. Pulse profile. Horizontal axis represents pixels. Vertical axis represents grey level.

3. Edge detection algorithm

The processing algorithm we have used is the edge detection algorithm proposed in Mallat and Zhong's work (Mallat & Zhong, 1992). For analysis purposes we have used six of the

eight possible scales for a 256x256 grey level image. The scale s forms a dyadic sequence (s = 2j, j = 1..6). The algorithm is summarized in figure 6, and table 1 shows the values of the normalization coefficients λ_j. The algorithm outputs are twelve 256x256 grey level images for each original one, six corresponding to the modulus of the derivative and six to the phase.

As we have pointed out in section 2, it is possible to perform a multiresolution analysis changing the smoothing function. The only difference between a typical Gaussian algorithm and the wavelet one is the filtering stage. In this case, instead of using the filter proposed by Mallat and Zhong, a Gaussian filter with a different size for each scale could be used. It leads to a different way to obtain the derivative of the image at different scales. A more detailed discussion about the Gaussian-based processing algorithm is given in Beltrán (Beltrán et al., 1998). Processing the image with the wavelet filter is faster, in computational cost terms, because the non-zero coefficients are constant independent on scale.

In order to obtain the contour image, a top-down searching algorithm has been implemented. For accepting a maximum at one scale to be an edge, it has been imposed that the maxima have to be propagated to the lowest scale with no change in the gradient direction between scales. When a maximum is found at one scale s we look for extrema in the lower scale within an interval of 2s centered in the extrema position at the scale s, in the direction given by the gradient. This interval is greater than the theoretical one found by Beltrán (Beltrán, 1994), in order to cover the maxima displacement.

The maximum appearing in the lower scale has to have the same direction that the upper one. The best edge position is given at the lowest scale. The first scale to be analyzed depends strongly on the image. Empirically we have noticed that this scale has to be no higher than either the 5th or the 6th. Otherwise, we have a strong blurring in the image that gives us information of global objects rather than finer patterns, like edges. The procedure has been iterated until first scale. A stop could be done at an upper scale, depending on the details we are looking for. If we were looking for finer details we should reach the first scale. A global threshold has been included in order to discard irrelevant edges.

j	λ_j
1	1.5
2	1.125
3	1.031
4	1.007
5	1.001
6	1

Table 1. Wavelet filtering normalization coefficients.

At the output of this block, we obtain a black and white image with the edge positions, and, for each detected edge, the values of the modulus of the gradient in each scale. The final step is the contour classification, which is made by analyzing the evolution of the modulus of the gradient across scales.
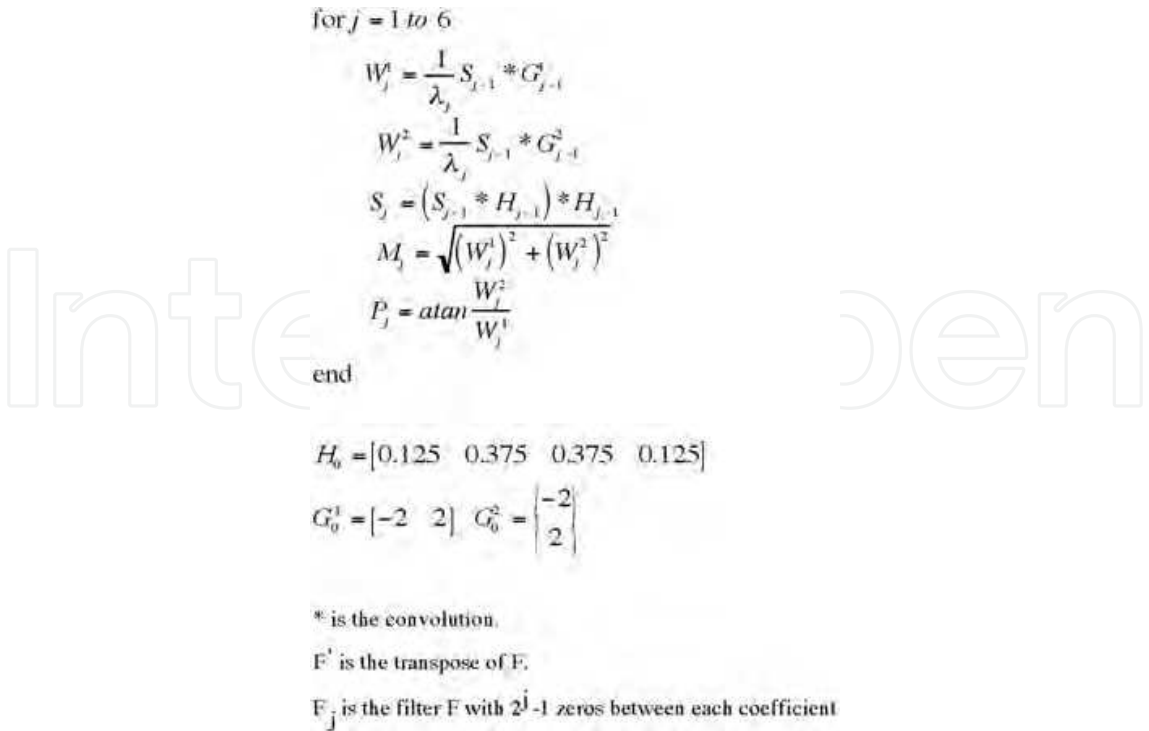
for $j = 1$ to 6

$$W_j^1 = \frac{1}{\lambda_j} S_{j-1} * G_{j-1}^1$$

$$W_j^2 = \frac{1}{\lambda_j} S_{j-1} * G_{j-1}^2$$

$$S_j = \left(S_{j-1} * H_{j-1} \right) * H_{j-1}'$$

$$M_j = \sqrt{\left(W_j^1 \right)^2 + \left(W_j^2 \right)^2}$$

$$P_j = atan \frac{W_j^2}{W_j^1}$$

end

$$H_0 = \begin{bmatrix} 0.125 & 0.375 & 0.375 & 0.125 \end{bmatrix}$$

$$G_0^1 = \begin{bmatrix} -2 & 2 \end{bmatrix} \quad G_0^2 = \begin{bmatrix} -2 \\ 2 \end{bmatrix}$$

* is the convolution.

F' is the transpose of F.

F_j is the filter F with $2^j - 1$ zeros between each coefficient

Fig. 6. Wavelet Algorithm.

4. Classification schema

To obtain real evolution patterns in an image we have analyzed the evolution across scales of the value of the wavelet transform modulus for each contour type at the point at which the edge is located in our test image (see figure 7).
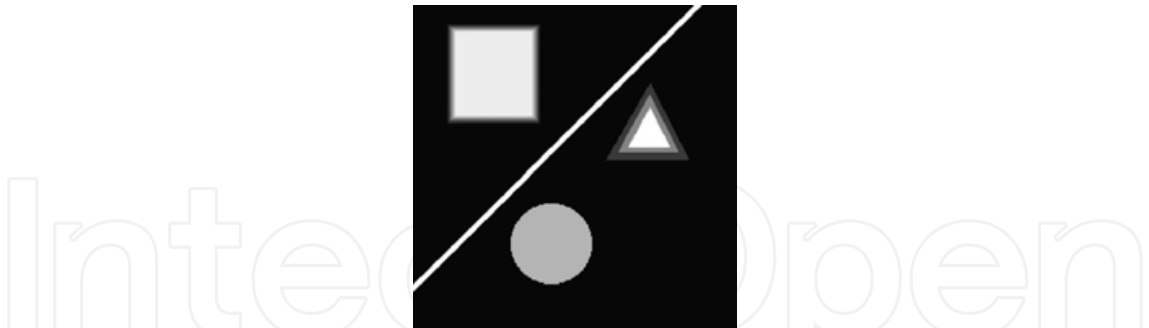
Fig. 7. Test image. Each object has a different contour profile. Circle: step. Square: ramp. Triangle: stair. Straight line: pulse.

In figure 8 we present for each contour class both the median value (normalized to the highest value for each pixel) and the deviation for each scale.

It can be seen a decreasing behavior in the evolution for each contour class in the upper scales due to the interaction between the opposite contours in the image. This decreasing pattern is not present in the 1-D case. From those results we can say that a step profile, figure 8(a), is characterized by an almost constant evolution across scales. The evolution pattern for a ramp, presented in figure 8(b), is a continuous growing from low to high scales. A constant value followed by an increasing in the normalized modulus of the gradient in the
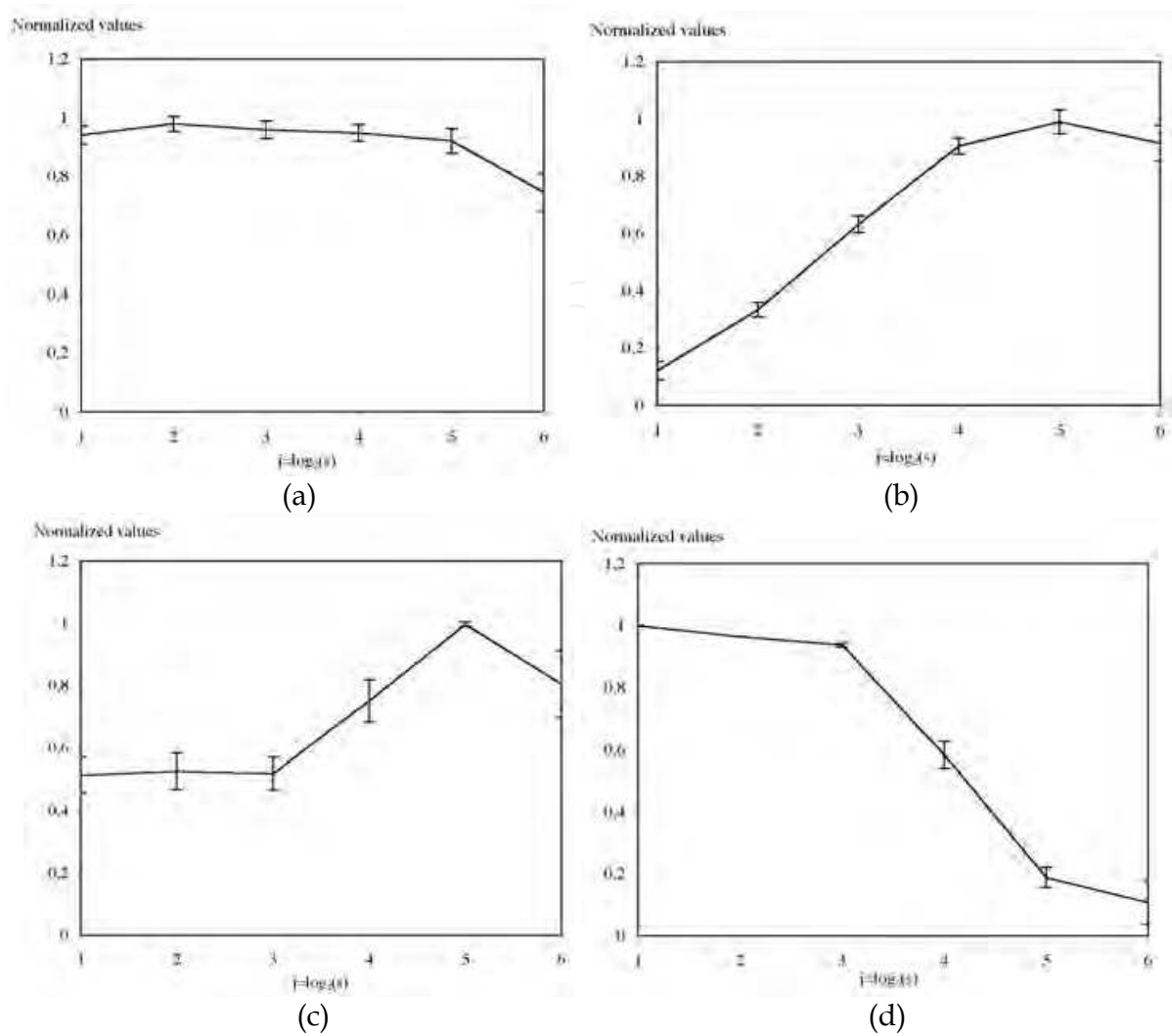
Fig. 8. (a) Step evolution. (b) Ramp evolution. (c) Stair evolution. (d) Pulse evolution. (See text for details)

edge position characterizes a stair contour, figure 8(c). As shown in figure 8(d) the pulse profile presents a sharp decreasing in the upper scales due to the interaction between the positive and negative slopes.

Figure 9 shows a block diagram of the edge detection and classification algorithm. An immediate conclusion is that it is possible to implement an algorithm to detect and classify the above-characterized four profiles using the wavelet transform coefficients. Then, we are able to distinguish one contour type from another one only by looking at the coefficients evolution at the appropriate contour point and with no pre-processing in the coefficient values.

The classification engine is based in a second-order polynomial-fitting algorithm. We have chosen a second order taking into account the easiness of implementation of properties such as derivability, continuity, concavity, and convexity. Analyzing the concavity and convexity, zero crossing and minimum abscissa and ordinate of the fitted polynomial we are able to distinguish between the four profiles presented. Typical values of the coefficients are presented in table 2 and the corresponding polynomial in figure 10. The second order polynomial is of the form: $f(x)=a_0+a_1x+a_2x^2$. We have used a standard polynomial-fitting algorithm with the 6 discrete values obtained for each contour at the edge location.
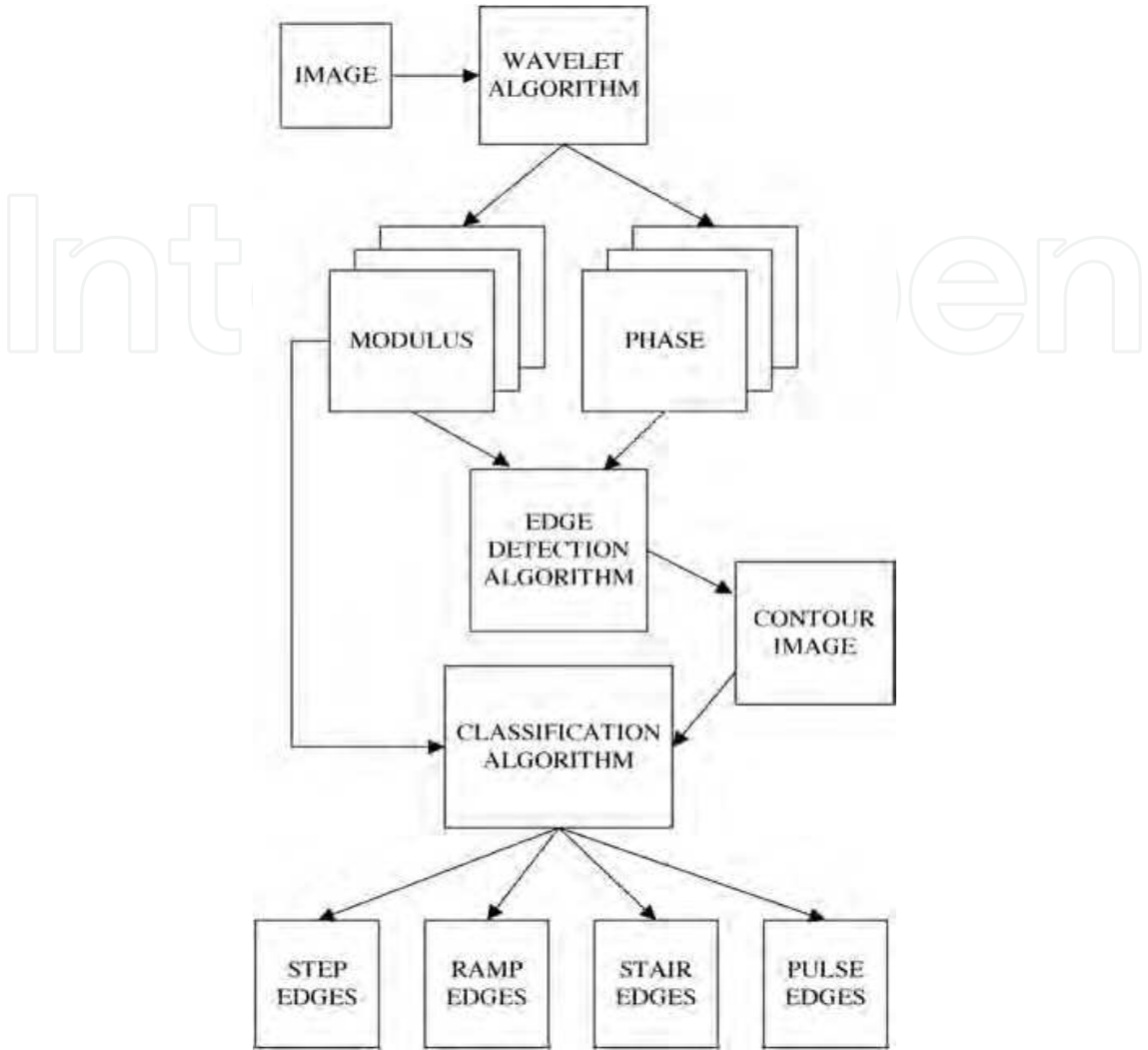
Fig. 9. Edge detection and classification algorithm.

	a_0	a_1	a_2
Step	0.8295	0.1026	-0.0187
Ramp	-0.0249	0.3678	-0.0362
Stair	0.3562	0.0894	0.0005
Pulse	1.0087	-0.0732	-0.0131

Table 2. Typical coefficients values of the fitted polynomial.

The decision algorithm is presented in figure 11. We first analyze the convexity of the polynomial (a_2>0), in the affirmative case we have a stair edge. In case of having a concave polynomial we analyze its slope. In case of a positive slope, we will have a ramp edge. In the opposite case, we will have to compute the zero crossing value. If this value is greater than, approximately, 6 the edge will be a step, otherwise it will be a pulse.
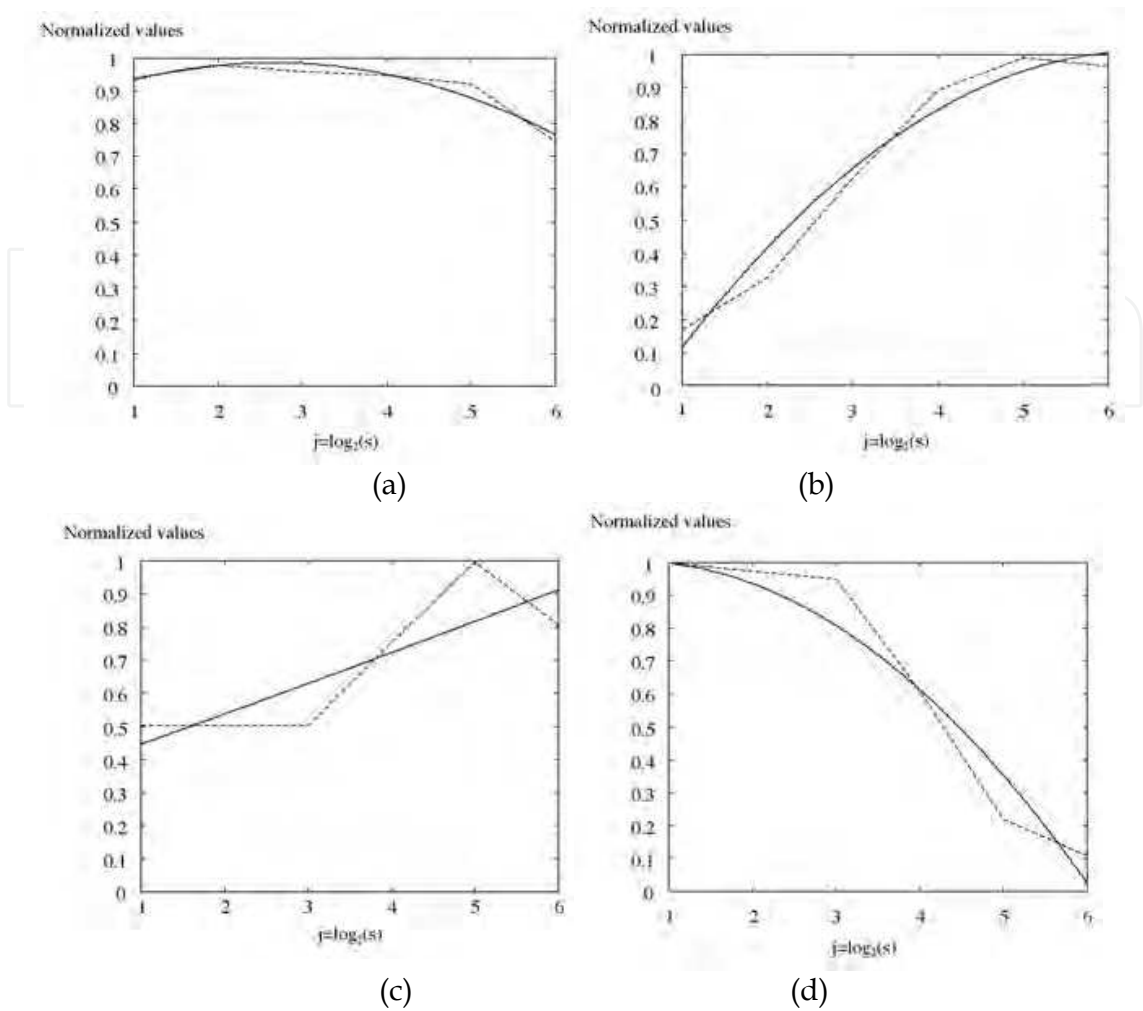
Fig. 10. Dotted lines: Original values. Solid lines: polynomial. (a) Polynomial fitting for the step profile. (b) Polynomial fitting for the ramp profile. (c) Polynomial fitting for the stair profile. (d) Polynomial fitting for the pulse profile.
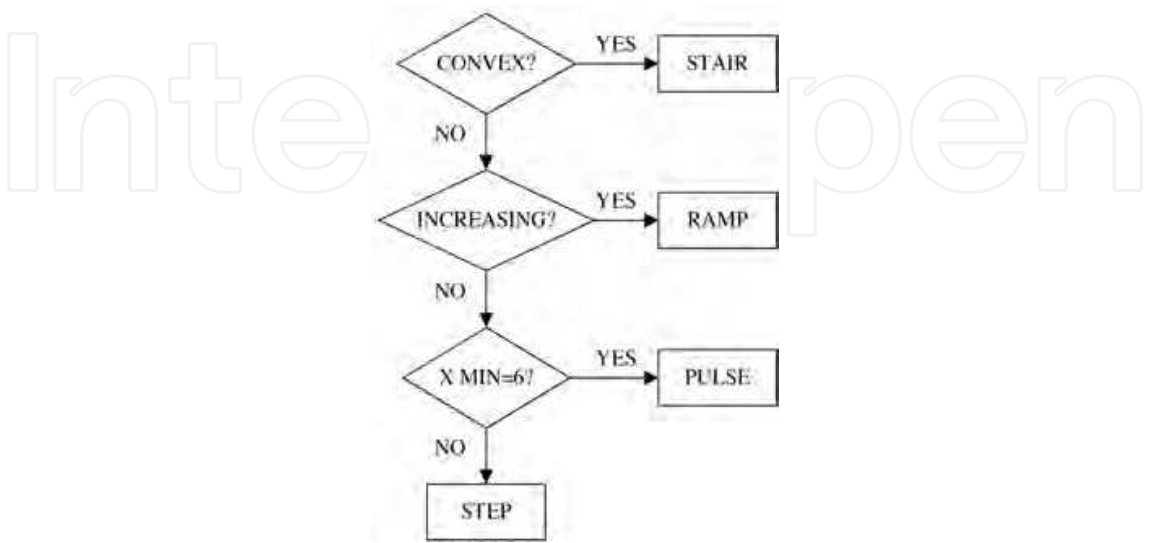
Fig. 11. Decision algorithm for edge classification.

As some preliminary results we can see the correct classification made in figure 12 for our test image. It is important to note that no post-processing (edge tracking, non-maxima suppression, etc.) has been made in the obtained contour image.
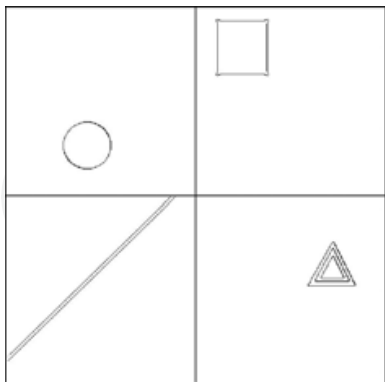
Fig. 12. From left to right and top to bottom: step, ramp, pulse, and stair classified profiles.

5. Noise characterization and polynomial classification

As a first practical result of our processing schema we have analyzed our algorithm behavior with respect to noise. In order to obtain a proper characterization of the noise we have to compare the evolution in the coefficients shown by the noise with that presented by other kind of profiles. Firstly, we have processed a simple image with only Gaussian noise. Secondly, we have analyzed the image with the edge detection and classification algorithm as described in section 4. The mean values of the modulus of the gradient evolution at the edge points are shown in figure 13a. This pattern characterizes isolated Gaussian noise.
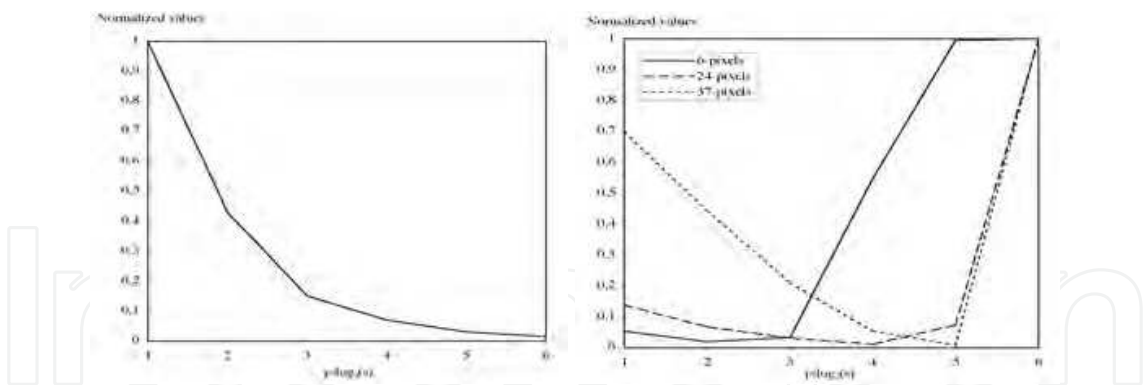

Fig. 13. (a) Mean values of Gaussian noise evolution. (b) Noise dependence on distance.

But noise is rarely presented isolated in a real image, and its evolution depends strongly on the distance between the noise and the contour of the objects present in the image. To obtain the noise evolution we have processed the image of a simple square with a step profile corrupted with Gaussian noise. The evolution in the values of the noise, in terms of the distance between the noise and the square, are presented in figure 13b. These patterns serve us to classify and differentiate the noise from other contour types. We can see that the evolution pattern for a noise contour located very close to a real contour is quite similar to the stair evolution. This is the expected behavior because the stair has been defined as closed steps. We can include this contour type in our classification engine.

The final decision algorithm is presented in figure 14. We have to distinguish between the stair type and the noise one. If we have a convex polynomial ($a_2 > 0$) we have a stair or noise edge. They are differentiated by means of the minimum ordinate value. If the minimum ordinate is close to 0 it is classified as a noise edge.
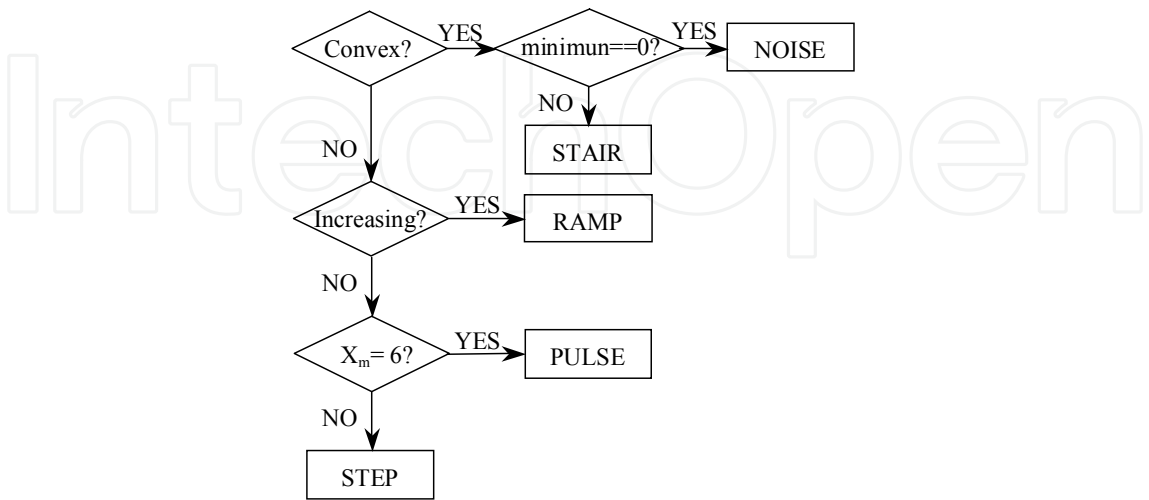
Fig. 14. Decision algorithm for edge classification.

To analyze the robustness of the algorithm we have corrupted our test image with 20 dB of Gaussian noise. The classification results are shown in figures 15 and 16, respectively. No thresholding has been applied to the contour image.
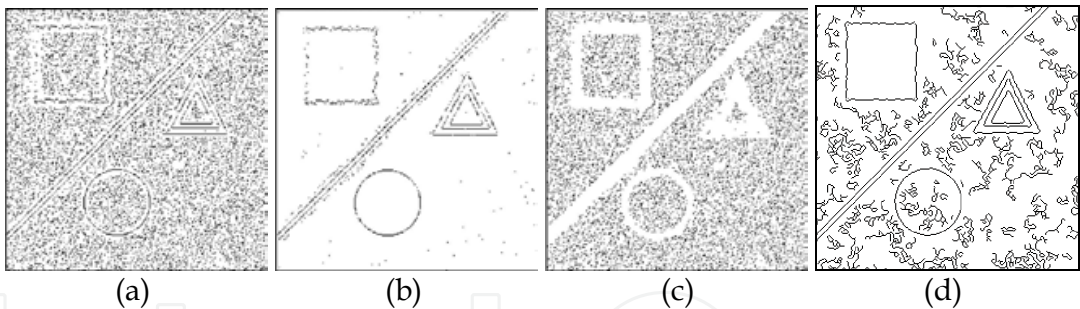
Fig. 15. Test image corrupted with 20 dB Gaussian noise. (a) Detected edges without thresholding. (b) Non noise edges. (c) Noise edges. (d) Detected edges with Canny algorithm.
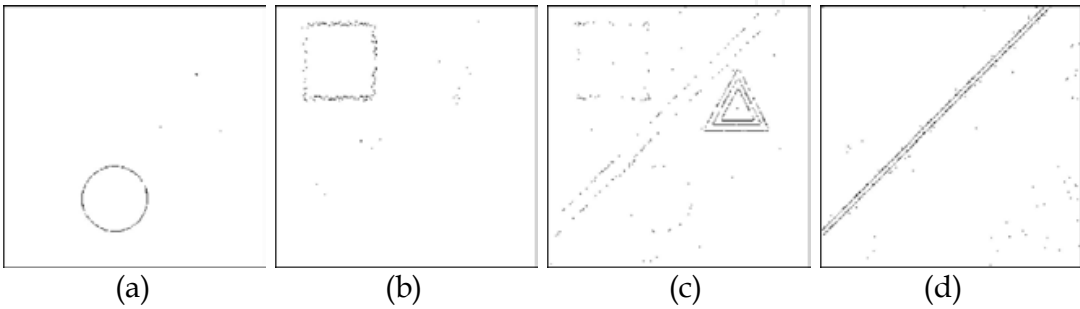
Fig. 16. Edge profiles detected for the test image corrupted with 20 dB Gaussian noise. (a) Step edges. (b) Ramp edges. (c) Stair edges. (d) Pulse edges.

It can be seen that the noise is perfectly classified (figure 15c) and can be eliminated by simply removing this edge type (figure 15b). Figure 15d shows the output of the Canny edge detector. In this case Canny operator is more sensitive to noise than our algorithm. Figure 16 shows the different edge profiles detected.

6. New contour types

The geometrical characterization of contour types gives us very promising results. To confirm the ability of our algorithm to distinguish different contour types we present here four new types of edge profiles that represent a more general gray-level transition than a simple step. We have included the roof profile, the ridge profile and two non-antisymmetrical step profile models that have been already considered previously by Paillou (Paillou, 1994).

6.1 Roof
A roof profile, named $R(x)$, is shown in figure 17. We have named the point x_0 the middle of the roof, ω is the roof width and U_0 the roof height.

$$r(x) = \begin{cases} 0 & \text{if } x < x_0 - \dfrac{\omega}{2} \\[2mm] mx + k & \text{if } x_0 - \dfrac{\omega}{2} \leq x \leq x_0 \\[2mm] U_0 & \text{if } x = x_0 \\[2mm] -mx + k & \text{if } x_0 \leq x \leq x_0 + \dfrac{\omega}{2} \\[2mm] 0 & \text{if } x > x_0 + \dfrac{\omega}{2} \end{cases} \tag{17}$$
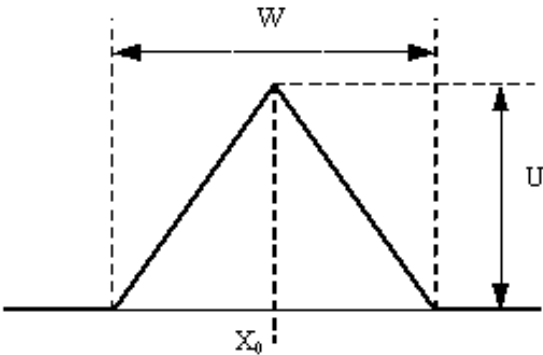

Fig. 17. Roof profile. Horizontal axis represents pixels. Vertical axis represents grey level.

6.2 Ridge
A ridge profile, named $R(x)$, is shown in figure 18. The point x_0 is the middle of the ridge, ω_1 is the width of the first ramp, ω_2 is the width of the plain part, while ω_3 is the width of the second ramp. U_0 is the ridge height.

$$
R(x) = \begin{cases}
0 & \text{if } x < x_0 - \omega_1 - \dfrac{\omega_2}{2} \\[2mm]
mx + k & \text{if } x_0 - \omega_1 - \dfrac{\omega_2}{2} \leq x \leq x_0 - \dfrac{\omega_2}{2} \\[2mm]
U_0 & \text{if } x_0 - \dfrac{\omega_2}{2} \leq x \leq x_0 + \dfrac{\omega_2}{2} \\[2mm]
-mx + k & \text{if } x_0 + \dfrac{\omega_2}{2} \leq x \leq x_0 + \dfrac{\omega_2}{2} + \omega_3 \\[2mm]
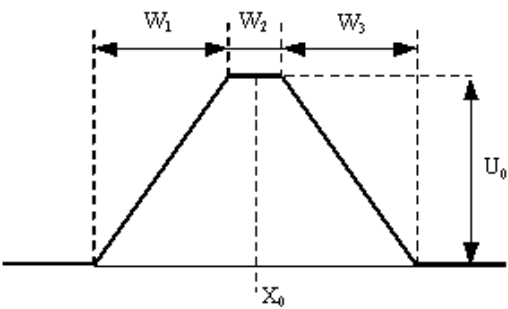0 & \text{if } x > x_0 + \dfrac{\omega_2}{2} + \omega_3
\end{cases}
\tag{18}
$$

Fig. 18. Ridge profile. Horizontal axis represents pixels. Vertical axis represents grey level.

6.3 First non-antisymmetrical step

The first non-antisymmetrical step profile, named nu(x), is shown in figure 19. The point x_0 is the step location, ω_1 is the width of the first ramp and ω_1 is the width of the second ramp. U_1 and U_3 are their respective heights. U_2 is the step height at position x_0. The first non-antisymmetrical step function can be written as:

$$
nu(x) = \begin{cases}
0 & \text{if } x < x_0 - \omega_1 \\
mx + k & \text{if } x_0 - \omega_1 \leq x \leq x_0 + \omega_1 - \omega_2 \\
mx + k & \text{if } x_0 + \omega_1 - \omega_2 \leq x \leq x_0 + \omega_1 + \omega_2 \\
U_3 & \text{if } x > x_0 + \omega_1 + \omega_2
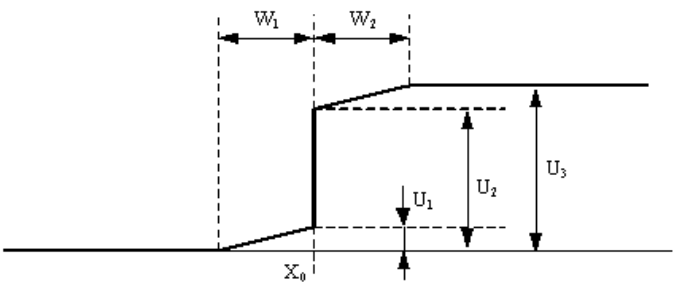\end{cases}
\tag{19}
$$

Fig. 19. First non-antisymmetrical step profile. Horizontal axis represents pixels. Vertical axis represents grey level.

6.4 Second non-antisymmetrical step

The second non-antisymmetrical step profile, named nu2(x), is shown in figure 20. The point x_0 is the middle of the main ramp, ω_1 is the width of the first ramp, ω_2 is the width of central ramp and ω_3 is the width of the third ramp, respectively. U_1, U_2 and U_3 are their respective heights (see figure). Let be nu2(x) the second non-antisymmetrical step function.

$$
nu2(x) = \begin{cases} 0 & \text{if } x < x_0 - \omega_1 \\[2mm] mx + k & \text{if } x_0 - \omega_1 \leq x \leq x_0 + \omega_1 - \dfrac{\omega_2}{2} \\[2mm] cx + k & \text{if } x_0 + \omega_1 - \dfrac{\omega_2}{2} \leq x \leq x_0 + \omega_1 + \dfrac{\omega_2}{2} \\[2mm] mx + k & \text{if } x_0 + \omega_1 + \dfrac{\omega_2}{2} - \omega_3 \leq x \leq x_0 + \omega_1 + \dfrac{\omega_2}{2} + \omega_3 \\[2mm] U_3 & \text{if } x > x_0 + \omega_1 + \dfrac{\omega_2}{2} + \omega_3 \end{cases}
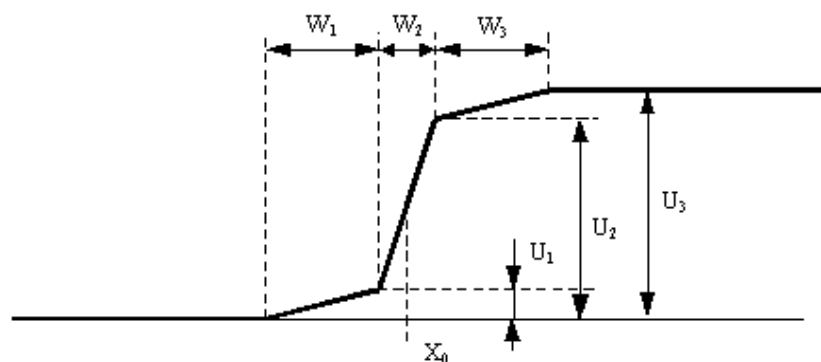\tag{20}
$$

Fig. 20. Second non-antisymmetrical step profile. Horizontal axis represents pixels. Vertical axis represents grey level.

6.5 Modified classification algorithm

Finally, we will include the four contour types in our classification algorithm. We will also work on the basis of the polynomial-fitting algorithm, by following characteristics such as convexity, concavity, zero crossings and minimum abscissa and ordinate as we did beforehand. Furthermore, we have also used a standard polynomial-fitting algorithm with the 6 discrete mean values obtained for each contour at the edge location. However, in this case, a third order polynomial fitting will serve us to classify some of the new profile models introduced above. In particular, the third order coefficient pattern matching will serve us to discriminate among the ridge, roof and ramp profiles, respectively. Maximum value at scales 5 or 6 and comparison between values at two different scales are also used. Figure 21 shows the complete decision algorithm for edge classification.
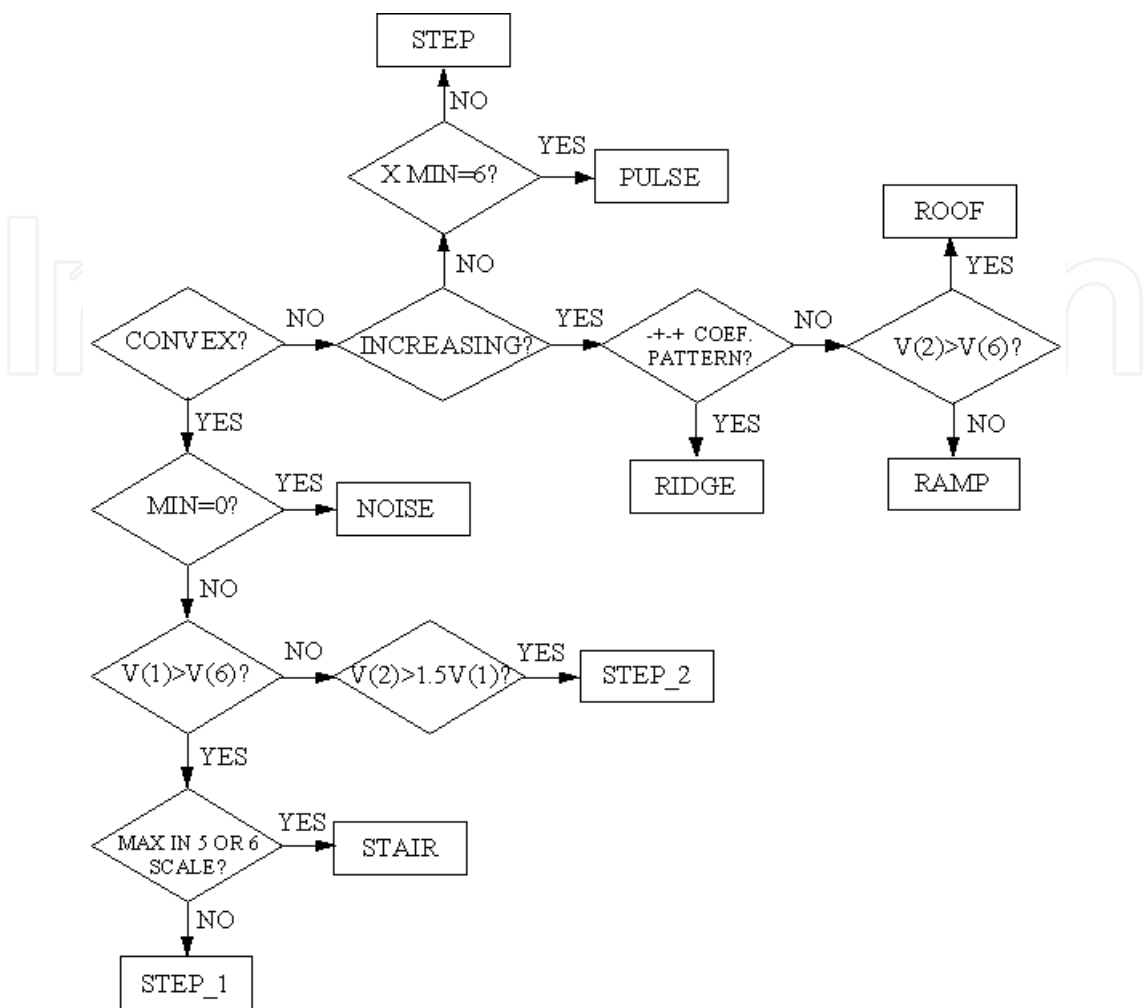
Fig. 21. Modified decision algorithm for edge classification.

The polynomial that characterizes the roof profile (as well as the ridge profile) is concave and with a positive slope in the first three scales, in the same way as the ramp profile does. In order to discriminate among these profile models just mentioned above, it is necessary to calculate the coefficient pattern for several fitting polynomials, third, fourth and fifth order). We find a remarkable difference among the ridge profile on one side, and the ramp and roof profiles on the other side. The coefficient pattern $(-,+,-,+) \equiv (a_0,a_1,a_2,a_3)$, characterizes the ridge profile. In order to distinguish between the ramp and roof profiles, we have realized that the mean value at scale 6 is greater that the one at scale 2 in a ramp profile, whereas both are similar in a roof profile. This is the condition to put aside both profiles.

As far as antisymmetrical step profiles are concerned, we must say that they both have some properties like the stair profile (convexity and minimum greater than zero). It can be shown that if the mean value at scale 6 is smaller than that at scale 1, we will obtain a stair profile, otherwise we will have the first non-antisymmetrical step profile. For the sake of algorithm efficiency, we have included a second condition: a stair profile has the maximum value at scales 5 or 6. In order to obtain the second non-antisymmetrical step profile, we compute whether the mean value at scale 2 is, at least 1.5 times the mean value at scale 1.

In order to show the results of the modified algorithm, we have created a new test image, including the new edge profiles. Figure 22 shows the new test image, together with the different detected edge profiles. Results with a real image are shown in figures 23 and 24.
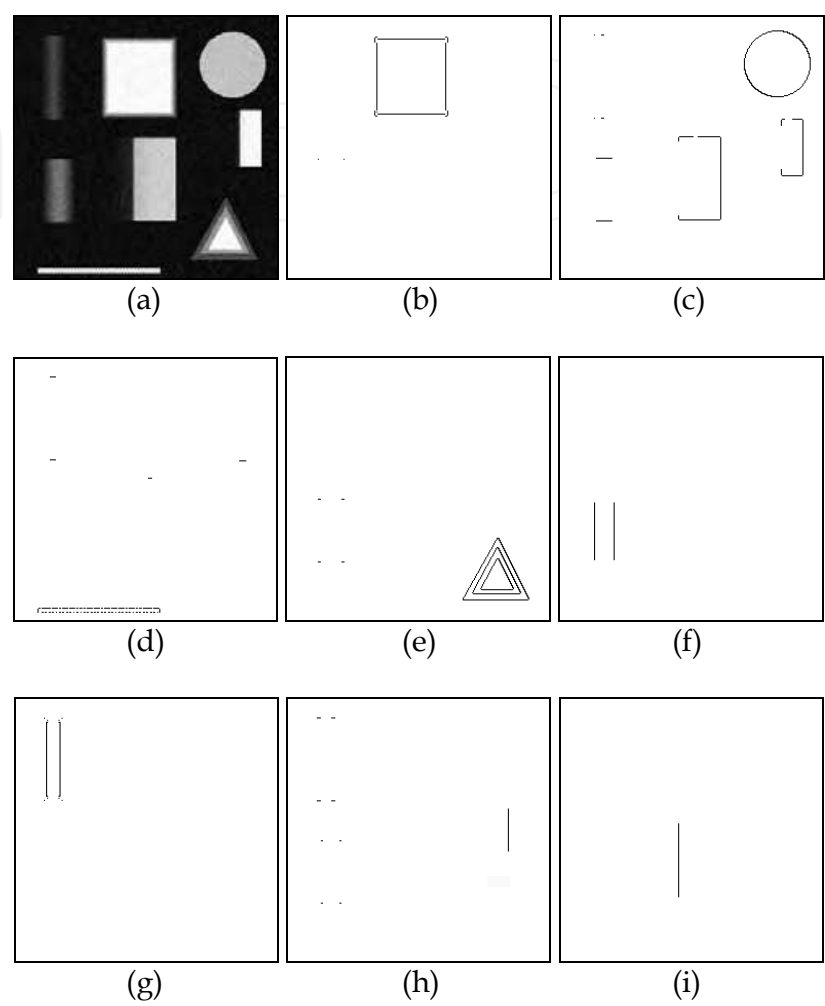

Fig. 22. (a) Test image. (b) Ramp edges. (c) Step edges. (d) Pulse edges. (e) Stair edges. (f) Ridge edges. (g) Roof edges. (h) First non-antisymmetrical step edges. (i) Second non-antisymmetrical step edges.
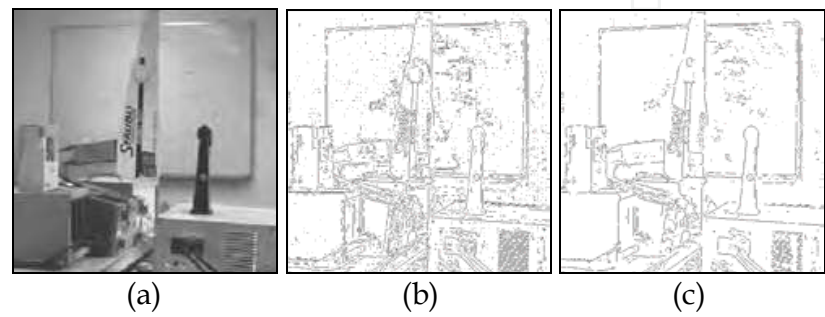

Fig. 23. (a) Original image. (b) Contour image. (c) Contour image after noise filtering.
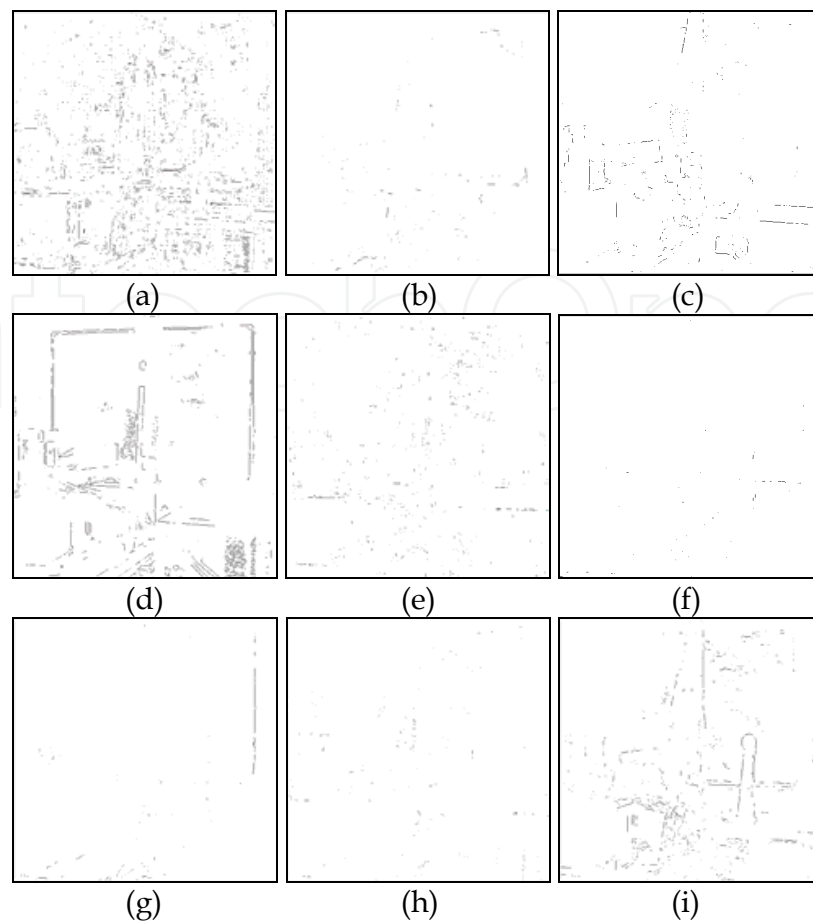
Fig. 24. (a) Noise edges. (b) Ramp edges. (c) Step edges. (d) Pulse edges. (e) Stair edges. (f) Ridge edges. (g) Roof edges. (h) First non-antisymmetrical step edges. (i) Second non-antisymmetrical step edges.

7. Conclusions

In this work, we have developed a new algorithm for edge detection and classification purposes using the coefficients given by the DWT. We have shown that Mallat's wavelet is a very suitable tool to perform both edge detection and contour analysis. We have presented the results obtained with a 256x256 synthetic image with several objects, each one with a different contour profile, obtaining a very good segmentation. At this point, we are not only able to see the evolution across scales of the edges proposed by Rosenfeld (Rosenfeld & Thurston, 1976) like in Williams and Shah's work (Williams & Shah, 1990, 1993), but we are also able to classify them.

A new edge class has been introduced: the noise. This edge type presents a particular evolution across scales. This has allowed us to implement a simply noise filtering algorithm based on edge classification.

The classification algorithm we have developed is based on second order polynomial fitting of the modulus of the wavelet transform coefficients. The mathematical behaviour of the polynomial is a robust indicator of the edge class. This kind of classification is good enough to obtain the five different profiles analyzed: step, ramp, stair, pulse, and noise. The robustness of the proposed classification schema has been tested including other profiles

appeared in the literature: roof, ridge and two kinds of non-antisymmetrical step models. A third order polynomial-fitting algorithm is needed to obtain a proper classification. This algorithm can be viewed as a new framework to classify different contour types.

Some preliminary results, like the behaviour of ramp edges, are promising to obtain a classification of the contours appearing in real images (shadows, changes in illumination, corners and so on). A future work to perform, which has not been covered in this paper, is the study of the evolution across scales of these real edges. If there were some special evolution pattern for these real edge types it would be very important information for the next stages of an image understanding algorithm.

An edge-closing algorithm based on the edge type is under developing in this moment. The extra information provided by the classification stage gives very good indicators to close edges and extract objects in an image. The processing results presented in this work have been obtained using Matlab®.

8. References

Beltrán, J. R., J. García-Lucía, J. & Navarro, J. (1994). Edge detection and classification using Mallat's wavelet. *Proceedings of the ICIP-94*, pp. 293-296

Beltrán, J. R., Beltrán, F & Estopañan A. (1998). Multiresolution edge classification: Noise Characterization, *Proceedings of the IEEE-SMC'98*, pp. 4476-448

Bergholm, F. (1987). Edge Focusing. *IEEE Trans. on Patt. Anal. and Machine Intell*, Vol. 9, No. 6, pp. 726-471

Canny, J. F. (1986). A computational approach to edge detection. I*EEE Trans. on Patt. Anal. And Machine Intell.* Vol. 8, pp. 679-698

Eklundth, J. O., Elfving, T. & Nyberg, S. (1982). Edge detection using the Marr-Hildreth operator whith different sizes. *Procdings of the. 6th. Int. Conf. on Pattern Recognition (ICPR),* Munich, Germany, pp. 1109-1112

Mallat, S. (1989). A theory for Multiresolution Signal Decomposition: The Wavelet Representation. *Trans. on Patt. Anal. and Machine Intell*, Vol. 11, No. 7, pp. 674-693

Mallat, S. & Zhong, S. (1992). Characterization of Signals from Multiscale Edges. *IEEE Trans. on Patt. Anal. and Machine Intell,* Vol. 14, No. 7, pp. 710-732

Marr, D. (1982). *Vision*. W. H. Freeman. San Francisco

Paillou, Ph. (1994). A non antisymmetrical edge profile detection. *Pattern Recognition Letters* Vol. 15, pp. 595-605

Park, D. J., Nam, Kwon M. & Park, Rae-Hong. (1995). Multiresolution Edge Detection Techniques, *Pattern Recognition,* Vol. 28, No. 2, pp. 211-229

Rosenfeld, A. & Thurston, M. (1971). Edge curve detection for visual scene analysis, *IEEE Transactions Computers,* Vol. 20, pp. 562-569

Rosenfeld, A. & Kak, A. C. (1976). Digital Picture Processing. Academic Press

Torre, V. & Poggio, T. (1984). *On edge detection*. Technical Report 768, MIT

Williams, D. J. & Shah, M. (1990). Normalized Edge Detector. *Proceedings of the 10th Int. Conference On Pattern Recognition,* 1 (16-21), pp. 942-946

Williams, D. J. & Shah, M. (1993). Edge Characterization Using Normalized Edge Detector. *Proceedings of CVGIP*, Vol. 5, No. 4, pp. 311-318

Witkin, A. P. (1983). Scale-space filtering. *Proceedings of the 4th Int. Joint Conference on Artificial Intelligence (IJCAI)*, pp. 1019-1022

Ziou, D. & Tabbone, S. (1993). A Multiescale Edge Detector. *Pattern Recognition*, Vol 26, No. 9, pp. 1305-1314

Discrete Wavelet Transforms - Algorithms and Applications

Edited by Prof. Hannu Olkkonen

The discrete wavelet transform (DWT) algorithms have a firm position in processing of signals in several areas of research and industry. As DWT provides both octave-scale frequency and spatial timing of the analyzed signal, it is constantly used to solve and treat more and more advanced problems. The present book: Discrete Wavelet Transforms: Algorithms and Applications reviews the recent progress in discrete wavelet transform algorithms and applications. The book covers a wide range of methods (e.g. lifting, shift invariance, multi-scale analysis) for constructing DWTs. The book chapters are organized into four major parts. Part I describes the progress in hardware implementations of the DWT algorithms. Applications include multitone modulation for ADSL and equalization techniques, a scalable architecture for FPGA-implementation, lifting based algorithm for VLSI implementation, comparison between DWT and FFT based OFDM and modified SPIHT codec. Part II addresses image processing algorithms such as multiresolution approach for edge detection, low bit rate image compression, low complexity implementation of CQF wavelets and compression of multi-component images. Part III focuses watermaking DWT algorithms. Finally, Part IV describes shift invariant DWTs, DC lossless property, DWT based analysis and estimation of colored noise and an application of the wavelet Galerkin method. The chapters of the present book consist of both tutorial and highly advanced material. Therefore, the book is intended to be a reference text for graduate students and researchers to obtain state-of-the-art knowledge on specific applications.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Guillermo Palacios, J. Ramón Beltrán and Raquel Lacuesta (2011). Multiresolution Approaches for Edge Detection and Classification Based on Discrete Wavelet Transform, Discrete Wavelet Transforms - Algorithms and Applications, Prof. Hannu Olkkonen (Ed.), ISBN: 978-953-307-482-5, InTech, Available from: http://www.intechopen.com/books/discrete-wavelet-transforms-algorithms-and-applications/multiresolution-approaches-for-edge-detection-and-classification-based-on-discrete-wavelet-transform

INTECH
open science | open minds

www.intechopen.com

Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

Phone: +86-21-62489820
Fax: +86-21-62489821